Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Canceled)

2. (Currently amended) A method for managing subsystem processes from a central site in a digital media distributor (DMD) system, the method comprising:

utilizing a plurality of threads, including utilizing a main manager thread, as a task manager and a main program thread of an operating system in a central site server of the [digital media distributor] (DMD) system; and

autonomously controlling initiation and termination of one or more subsystem processes associated with data object transmissions of the [digital media distributor] DMD system with the task manager, wherein autonomously controlling further comprises managing subsystem processes associated with data object transmissions of the DMD system from start-up to shut down, including states of online, offline, process inoperable, deadlock inoperable, and spawn inoperable, to dynamically manage the DMD system.

3. (Original) The method of claim 2 wherein utilizing a plurality of threads further comprises utilizing a subsystem control thread as a child thread of the main manager thread.

4. (Original) The method of claim 3 wherein utilizing a plurality of threads further comprises utilizing a watchdog worker thread as a child thread of the subsystem control thread.

5. (Original) The method of claim 4 wherein utilizing a plurality of threads further comprises utilizing a spawn worker thread as a child thread of the watchdog worker thread.

6. (Original) The method of claim 5 further comprising utilizing one watchdog worker thread and one spawn worker thread for each subsystem process.

7. (Original) The method of claim 3 further comprising utilizing the subsystem control thread to determine need for initiation of a subsystem process.

8. (Original) The method of claim 6 further comprising utilizing the watchdog worker thread to start each subsystem process through the spawn worker thread and to monitor performance of each subsystem process.

9. (Original) The method of claim 8 wherein utilizing the spawn worker thread further comprises spawning each subsystem process and waiting for termination of each spawned subsystem process.

10. (Currently amended) A method for managing subsystem processes from a central site in a digital media distributor system, the method comprising:

utilizing a plurality of threads, including utilizing a main manager thread, as a task manager in a central site server of the digital media distributor system; and

autonomously controlling initiation and termination of one or more subsystem processes associated with data object transmissions of the digital media distributor system with the task manager, including controlling a subsystem process from the group comprising a scheduler process, a stage manager process, a local insertion system proxy process, an error document check process, a response document processor process, a disk pool manager process, a request generator process, As-Run manager processes, an update network break time process, and a network local broadcast process.

11. (Canceled)

12. (Currently amended) A digital media [distribution] distributor (DMD) system with centralized management of subsystem processes, the DMD system comprising:

a distribution network for data object transmission;

a central site server, the central site server utilizing a main manager thread for a task manager as a main program thread of an operating system for autonomous control of initiation and termination of one or more subsystem processes associated with data object transmission of the DMD system, wherein the one or more subsystem processes associated with data object transmission of the DMD system further comprise subsystem

processes from start-up to shut down, including states of online, offline, process inoperable, deadlock inoperable, and spawn inoperable; and

a plurality of remote site servers for receiving data object transmissions from the central site server via the distribution network.

13. (Previously presented) The system of claim 12 wherein the central site server utilizes a subsystem control thread as a child thread of the main manager thread.

14. (Previously presented) The system of claim 13 wherein the central site server utilizes a watchdog worker thread as a child thread of the subsystem control thread.

15. (Previously presented) The system of claim 14 wherein the central site server utilizes a spawn worker thread as a child thread of the watchdog worker thread.

16. (Previously presented) The system of claim 15 wherein the central site server utilizes one watchdog worker thread and one spawn worker thread for each subsystem process.

17. (Previously presented) The system of claim 13 wherein the central site server further utilizes the subsystem control thread to determine need for initiation of a subsystem process.

18. (Previously presented) The system of claim 16 wherein the central site server further utilizes the watchdog worker thread to start each subsystem process through the spawn worker thread and to monitor performance of each subsystem process.

19. (Previously presented) The system of claim 18 wherein the central site server further utilizes the spawn worker thread for spawning each subsystem process and waiting for termination of each spawned subsystem process.

20. (Previously presented) A computer readable medium containing program instructions for managing subsystem processes from a central site in a digital media distributor (DMD) system, the program instructions comprising:

providing a task manager as a main program thread of an operating system of a central site server of the DMD system; and

managing subsystem processes associated with data object transmissions of the DMD system from start-up to shut down, including states of online, offline, process inoperable, deadlock inoperable, and spawn inoperable, with the task manager to dynamically manage the DMD system.

21. (Previously presented) The program instructions of claim 20 wherein providing a task manager further comprises utilizing a control thread and worker threads for managing the subsystem processes.

22. (Canceled)

23. (Canceled)